REMARKS

Claims 1-21 remain in the application. Claims 1 – 21 have been rejected. Claims 1-5, 8-11, 13-16, and 19-21 have been amended. Applicant respectfully requests reconsideration of the pending claims.

CLAIM REJECTIONS UNDER 35 USC §112

The Office Action rejected claim 1 under 35 USC 112, second paragraph, as being indefinite. Claim 1 has been amended to clarify the term "document" so as to clearly distinguish it as the electronic document which is the subject of the claim.

CLAIM REJECTIONS UNDER 35 USC §102

The Office Action has rejected claims 1, 3, 5, 6, 8-11, 12, 14, 16, 17, and 19-21 under 35 USC 102(e) as being anticipated by Lindblad et al., US Patent Number 7,171,404. Applicant respectfully disagrees.

As to claims 1, 11, and 21, Lindblad does not teach or suggest the following claimed elements of claims, 1, 11, and 21:

wherein the search criteria comprise a set of constraints that specify forward *or* backward relations between nodes;

receiving a context node from the electronic document with respect to which the search criteria are applied;

modifying the search criteria to introduce a constraint matching the context node into the set of constraints;

processing the electronic document in a *streaming* manner and using the modified search criteria; and

locating one or more nodes that satisfy the modified search criteria.

For a reference to anticipate a claim, each element and limitation of the claim must be found in the reference. Hoover Group, Inc. v. Custom Metalcraft, Inc., 66 F.3d 299, 302 (Fed. Cir. 1995). Lindblad teaches "methods for generating a pre-computed index that is used for generating a result for a query. Step queries are pre-computed and the results to these step queries generated and stored in the index along with the step queries" See Lindblad [0226]. Lindblad follows the following steps: 1) computing relationships among elements in a document; 2) generating step queries from relationships; 3) reducing step queries to their canonical form; 4) generating a hash key for each canonical form; 5) generating results for step queries; and 6) storing step query results and hash keys in a database. See Lindblad [Fig. 10, paragraphs [0046] – [0102]]. Lindblad does not teach or suggest using search criteria comprising constraints that specify forward *or* backward relations between nodes. "[00129] A search for a given subtree id proceeds by scanning down the list"

Lindblad does not teach or suggest applying search criteria to a context node from the electronic document. Instead, Lindblad teaches "receiving the query, wherein the query comprises a path of elements in the document of elements…" Lindblad, Claim 1.

Lindblad does not teach or suggest modifying the search criteria to introduce a constraint matching the context node into the set of constraints; rather Lindblad teaches "reducing the step query into a plurality of step queries" and "for each of the plurality of step queries, retrieving a pre-computed step query result for a step query in the plurality of step queries by querying the database using the step query" Lindblad, claim 1.

Lindblad does not teach or suggest processing the electronic document in a streaming manner. At page 5 of the Office Action, paragraph 8, the Examiner states: "however he [Lindblad] does not explicitly state that the document is stored in memory or that the document is streaming"

Claims 1, 11, and 21 are clearly not anticipated by Lindblad; therefore they should be allowed.

As to claims 3 and 14, these claims are dependent on claims 1 and 11, respectively; therefore they are not allowable for at least the same reasons that their parent claims are allowable.

As to claims 5, 12, and 16 the Office Action contends that Lindblad indirectly discloses modifying the search criteria such that constraints specifying a backward relation may be reformulated into forward constraints. The Office Action is no doubt referring to the "post-processing" step S512 of Fig. 11. This step involves "[0176] For each node c labeled C, generate the ancestor path going back two steps, and check if grandparent(c) = A, parent(c) = B. If not, discard c, and loop around to process the next node." This step is an optional step performed *after* the query result is generated. This is not a modification of the search criteria, as in the instant application. In the instant application, the search criteria are modified so that constraints specifying a backward relation (such as "if given X, find parent Y") are reformulated into forward constraints, using an XDAG. Lindblad does not modify his search criteria in this manner; rather, Lindblad reduces his query into step queries, with each step representing levels.

Further, in Lindblad, by the time the optional post-processing step is performed, the document has already been traversed *at least* once. The instant application uses a method to guarantee a single traversal of the document. See the instant application at paragraph [0027]: "[0027] Current algorithms would evaluate this XPath expression by finding each "b" descendant of "X" and for each of these "b" elements, traverse up the tree until an "a" element is reached, and then, search all subtrees rooted at the "a" elements for "d" descendants. Note that elements in the subtrees rooted at "a" (subtrees 220, 224, and 222 and elements 206 and 208) will be visited several times, once for each "b" descendant of "X." Our algorithm will perform a single depth-first traversal of the input document 107 to determine all "b" elements that are descendants of the context node "X" such that all the abovementioned conditions specified by the XPath are satisfied."

As to claims 6 and 17, these are dependent upon independent claims 1 and 11, respectively, and therefore allowable for at least the same reasons that their parent claims are allowable.

As to claims 8-10, 19 and 20, Lindblad does not teach or suggest reordering a tree structure such that the number of nodes traversed is minimized. Paragraph 105 of Lindblad refers to a Posting List which is another document created by Lindblad's method in order to store step query results. Therefore, claims 8-10, 19 and 20 are not anticipated by Lindblad and are therefore allowable.

CLAIM REJECTIONS UNDER 35 USC §103

The Office Action rejected claims 2, 4, 13 and 15 under 35 USC(a) as being unpatentable over Lindblad in view of Raboczi et al. (US Publication 2003/0074352). The Office Action states that although Lindblad does not teach working with a "streaming" document, Raboczi teaches a database where the user can access a "streaming" document. Applicant respectfully disagrees. Raboczi teaches a database query system; however, Raboczi's system does not deal with streaming data at all. The Office Action correctly states that Raboczi's system stores a document on a memory device or allows internet access. However, the Office Action confuses internet access of a document with "streaming." A streaming document is an electronic document that can be viewed in real-time without first downloading (or buffering) the entire file. An example of a streaming document is the stock market ticker tape. Raboczi does not teach or suggest such a document.

The Office Action has rejected claims 7 and 18 under 35 USC 103(a) as being unpatentable over Lindblad et al. in view of Trappen et al. (US Publication 2005/0138064). The Office Action states that Lindblad does not teach using a modified directed acyclic graph comprising a node "CTXT" which only matches the context node, but that Trappen teaches the use of a DAG for a query search. Trappen deals with primarily SQL database queries and employs a DAG structure. However; this is not the same as using the modified DAG structure as in the instant application. In the instant application, an X-DAG is used. An X-DAG is a graph representation of an internal data structure created from an XPath expression. It represents the XPath expression and all of its constraints. The use of the X-DAG in the instant application is used to represent the data structure in such a way that backward constraints are represented as forward constraints. This is a far cry from Trappen's conventional use of the DAG to represent only forward constraints.

The Office Action further contends that it would have been obvious to make the above combination "because this allows for much faster search than the traditional text (serial) query." Office Action at page 6. "When a rejection depends on a combination of prior art references, there must be some teaching, suggestion, or motivation to combine the references." In re Rouffet, 149 F.3d 1350, 1355, 47 USPQ2d 1453, 1456 (Fed. Cir. 1998) (citing In re Geiger, 815 F.2d 686, 688, 2 USPQ2d 1276, 1278 (Fed. Cir. 1987)). "Obviousness cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching or suggestion supporting the combination." ACS Hosp. Sys., Inc. v. Montefiore Hosp., 732 F.2d 1572, 1577, 221 USPQ 929, 933 (Fed. Cir. 1984). The showing must be clear and particular. Broad conclusory statements regarding the teaching of multiple references standing alone are not evidence. In re Dembiczak, 175 F.3d 994 (Fed. Cir. 1998).

Trappen does not teach using an X-DAG comprising a specific context node. In the instant case the showing for the motivation is not clear or specific and therefore the rejection is improper because of an improper use of hindsight.

For the foregoing reasons, Applicant respectfully requests allowance of the pending claims.

Respectfully submitted,

Michael J. Buchenhorner

Michael J. Buchenhorner
Reg. No. 33,162

Date: February 22, 2007

Michael Buchenhorner, P.A.
8540 S.W. 83 Street
Miami, Florida 33143
(305) 273-8007 (voice)
(305) 595-9579 (fax)